

Founder's Syndrome

The simplest way to do something, of course, is to do it yourself. But there's lots of stuff to be done and not enough you to do it. You can get around this a little bit by finding friends who are interested in doing some of it themselves, but at some point you're going to have to start "delegating", or getting somebody else to do it.

Now you're moving up in the world. If you're a decent manager, instead of running one project, you can run five or ten. Instead of simply directing your own labor, you can direct whole groups of people. Of course, it'll still be people doing what you wanted, but — funny thing — people are smart enough that they'll begin to get the gist of things on their own and, if you do it right, your delegation will become an organization. You can disappear for a week and things will keep on marching.

But make no mistake, even those organizations are still following the will of their erstwhile founder. Even as they get big, they betray facets of the founder's personality. The most obvious is in who is respected. My friend Emmett Shear has a theory that in each company only one class of people can be in charge and it's going to be the class of people the founders are in. At Apple, for example, the UI designers are in charge, because Jobs obsesses over UI design. At Google, it's the programmers, because Larry and Sergey used to code. Even though the founders aren't directly involved in every project, their surrogates still win the day.

Now the problem comes when the organization wants to grow beyond its founder. This is most common on non-profits, where they even have a name for it: founder's syndrome. See, once you have all these people carrying out your bidding, it's pretty difficult to want to give that up. Maybe you can have them do more projects, maybe you can give them more flexibility in what they choose, but I can't think of a single story where the guy in charge voluntarily gave up his power. And that has a severe cost (which I've come to calling the "power premium") because giving up your power is often the right thing to do.

In non-profits, for example, your organization is probably made up of a bunch of independent-minded young people with a strong belief in democracy. These people aren't too happy being told what to do all the time, especially when the instructions are pretty obviously not the best thing for the non-profit's mission. So they rebel against the founder, and the founder tries to hold on to power, and things get very messy. (I don't know how things usually turn out in this situation. Maybe you fight until the founder dies?)

Less well noticed is that the same mistake is made by for-profit corporations as well, it's just less obvious because the founder is the fellow holding all the cash, so you fight about it at your peril. But companies regularly do stupid things, even when if you asked all the people in the company about it they would have told you they were stupid. But in a capitalist economy, the founder has to maintain control.

But the power premium has a even more serious cost. While many people seem to be able to make the leap from doing something themselves to building an organization to do it, nobody seems to have been very good at taking the next step: going from an organization to a meta-organization, an organization that hires other organizations to do its work, rather than hiring people directly.

Why? Partly because few people get to be in charge of something the size of Google, where they need to take that next step to grow, and perhaps the trait of thinking that big is rare. But I think part of it is simply because the people at the top can't give up their power. Engaging organizations means you're no longer in charge of what people do or how they do it; the organizations have to be in charge of that. And that means you're no longer a delegator, but more of a moderator. It's founder's syndrome at the largest scale.

You should follow me on twitter [here](#).

October 27, 2006